Learning Hidden Markov Models for Anomaly Detection in Time Series

Kareth León\(^1\)  Henry Arguello\(^1\)  Jean-Yves Tourneret\(^2\)
Collaborators: Florian Mouret, TerraNIS

\(^1\)Universidad Industrial de Santander, Colombia
\(^2\)University of Toulouse, IRIT/INP-ENSEEIHT/TeSA, France

March 2020
A time series is a set of points indexed by time.

Temporal features extracted from high-dimensional signals indexed by time (e.g. spectral video, multitemporal hyperspectral images)
Context: Time Series

A time series is a set of points indexed by time.

Temporal features extracted from high-dimensional signals indexed by time (e.g. spectral video, multitemporal hyperspectral images)

Goal: Learn the temporal structure to discriminate anomalies!
What is an anomaly?
Some Applications

- **Networks and Communication**: Network Intrusion, Malware Detection, among others.
- **Medical**: Heart rate (ECG)
- **Economy**: Fraud Detection
- **Agriculture**: Crop Monitoring
- **Seismology**: Seismic Activity

Networks and Communication: Network Intrusion, Malware Detection, among others.
Anomaly Detection: Type of Anomalies

**Point Anomalies**
- An individual data instance is anomalous

**Contextual Anomalies**
- A data instance is anomalous in a specific context (but not otherwise).

**Collective Anomalies**
- A collection of related data instances is anomalous with respect to the entire data set, but not individual values (e.g. breaking rhythm in ECG)
Markov Model (first-order): Stochastic model for changing systems.

For the sequence \( \{z_1, \ldots, z_T\} \): 
\[ P(z_{t+1} | z_t, z_{t-1}, \ldots, z_1) = P(z_{t+1} | z_t) \] (Rule).

States \( S = \{\text{healthy, stressed}\} \).

Ex: Given that today the plant is healthy (\( z_t = \text{healthy} \)), the probability that it can be stressed tomorrow \( z_{t+1} \) is:
\[ P(z_{\text{today}} | z_{\text{tomorrow}}) = P(z_{\text{today}} = \text{healthy} | z_{\text{tomorrow}} = \text{stressed}) = 0.2. \]

Hypothetical situation
Suppose that you are allergic to plants so you cannot check the state of the plant. The only way to know the state of the plant is by observing if the caretaker of the garden lets a bottle of water or not in the kitchen.
Thus, instead of observing the real state, you observe the bottle. The real state of the plant is hidden!

Components

1. States: \( S = \{\text{healthy, stressed}\} \)
2. Observations: \( x = x_1, x_2, ..., x_T \)
3. Initial probabilities: \( \pi \)
4. Transitions probabilities: \( A \)
5. Emission probabilities: \( B \)

Emission probabilities

<table>
<thead>
<tr>
<th>Probability of observation</th>
<th>Bottle</th>
<th>Crossed Bottle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stressed</td>
<td>0.1</td>
<td>0.9</td>
</tr>
<tr>
<td>Healthy</td>
<td>0.8</td>
<td>0.2</td>
</tr>
</tbody>
</table>
Hidden Markov Models

- Observed time series: \( x = [x_1, ..., x_T] \),
- Set of possible states: \( S = \{s_1, ..., s_D\} \),
- Hidden sequence: \( z = [z_1, ..., z_T] \),
- Number of states: \( D \).

A HMM model is given by \( \theta = \{\pi, A, B\} \).

**Initial Probabilities:** \( \pi \)

\[
\pi_i = P(z_1 = s_i), \quad i = 1, ..., D
\]

**Transition Probabilities:** \( A \)

Probability to go from state \( i \) to state \( j \):

\[
a_{i,j} = P(z_{t+1} = s_i | z_t = s_j)
\]

where \( \{s_i, s_j\} \in S \), and \( i, j \in \{1, ..., D\} \).

**Emission Probabilities:** \( B \)

Observation probability distribution in state \( i \) such that \( B = \{b_i(\cdot)\} \). The probability of the observations can be:

- Discrete
- Continuous
Observation Probability Distribution $B = \{b_i(\cdot)\}$

- **Discrete Observations:** Observations can belong to a codebook $V = \{v_1, \ldots, v_K\}$, e.g., for the bottle observation Y (yes) or N (not), the codebook is $V = \{Y, N\}$.

  The probability is defined as:

  $$b_i(x_t) = P(x_t = v_k | z_t = s_i), \quad (1)$$

  where $1 \leq i \leq D$, $t = 1, \ldots, T$.

- **Continuous Distribution:** Observations follow a specific distribution, e.g., a Gaussian distribution or the mixture of multiple Gaussians.

  The emission probabilities are defined as:

  $$b_i(x_t) = \sum_{m=1}^{M} C_{i,m} \mathcal{N}(x_t | \mu_{i,m}, \Sigma_{i,m}), \quad (2)$$

  where $1 \leq i \leq D$, $M$ is the number of Gaussians, $\mu$ is the mean, $\Sigma$ is the covariance, and $\sum_{m=1}^{M} C_{i,m} = 1$. 

  ![Continuous Distribution Diagram](diagram.png)
In general, there exists three main tasks related to the HMMs:

1. **Evaluation**: Given a HMM model $\theta$ and $x$, estimate the probability of observation: Find $P(x|\theta)$.
   
   Solution:
   
   $P(x|\theta) = \sum_{all z} P(x, z|\theta)$. Forward-Backward Algorithm.

2. **Decoding**: Given a HMM model $\theta$ and observed sequence $x$, compute the hidden sequence that best models the observations: Find $z$.
   
   Solution:
   
   $P(x, z|\theta) = P(x|z, \theta)P(z, \theta)$. Viterbi Algorithm.

3. **Learning**: Given the observed sequence $x$, estimate the most likely HMM model $\theta$ using the maximum likelihood method: Find $\theta$.
   
   Solution:
   
   Choose $\theta = \{\pi, A, B\}$ such that $P(x|\theta)$ is maximized. Maximum likelihood estimation.
The proposed approach has two steps: Learning and Testing.

**Learning**
- Set of Nominal Signals
- Nominal Model Estimation
  - Find the model $\theta_n$ for each $x_n$: $\hat{\theta}_n = \arg\max_{\theta} P(x_n | \theta)$
- Model Selection
  - $\Theta = \{\hat{\theta}_1, \ldots, \hat{\theta}_L\}$

**Testing**
- Set of test Signals
- Probability Estimation Matrix
  - $W_{m,\ell} = P(x_m | \hat{\theta}_\ell)$
- Score Vector
  - $w_m = \max \{W_{m,\ell}\}_{\ell=1}^L$
- Detection
  - $w_m \overset{H_0}{\overset{H_1}{\sim}} \alpha$
  - Estimated Labels

**Hypothesis**

The following binary hypothesis test is considered to discriminate anomalies:

- $H_0 :$ Absence of anomalies
- $H_1 :$ Presence of anomalies
Let \( X = [x_1, ..., x_N]^T \) be a set of time series, where \( x_n = [x_{n,1}, ..., x_{n,T}] \), with \( x_n \in \mathbb{R}^T \).

Learning

- Learn the HMM model parameters \( \theta_n = \{\pi^{(n)}, A^{(n)}, B^{(n)}\} \) that maximize the log-probability of the observed sequence \( x_n \):

\[
\tilde{\theta}_n = \arg \max_{\theta} \log(P(x_n|\theta)), \ n = 1, ..., N
\]  

- \( x_n \): \( n \)-th time signal from the set \( X \in \mathbb{R}^{N \times T} \)
- Efficiently solved via the Baum-Welch algorithm
- The set of learned HMM is written as \( \Theta = \{\tilde{\theta}_1, ..., \tilde{\theta}_N\} \)
- Select \( L \) HMM models for the testing as: \( \{\tilde{\theta}_{\ell_1}, ..., \tilde{\theta}_{\ell_L}\} \), with \( L \ll N \).
HMM-Learning for Anomaly Detection

Set of test Signals
\[ \mathbf{X} \in \mathbb{R}^{M \times T} \]

Probability Estimation Matrix
\[ \mathbf{W}_{m,\ell} = P(x_m|\theta_\ell) \]

Score Vector
\[ w_m = \max \{ \mathbf{W}_{m,\ell} \}_{\ell=1}^L \]

Detection
\[ \mathbf{W}_{m,\ell} \overset{H_1}{\bowtie} \alpha \]

Testing

- Estimate the probability of observation of \( x_m \) by using \( \{\tilde{\theta}_1, ..., \tilde{\theta}_L\} \):
  \[ \mathbf{W}_{m,\ell} = P(x_m|\tilde{\theta}_\ell), \]  \( (4) \)

  - \( x_m \): \( m \)-th test signal, with \( m = 1, ..., M \).
  - \( \mathbf{W}_{m,\ell} \): matrix of probabilities of test sequences, with \( \ell = 1, ..., L \), \( \mathbf{W} \in \mathbb{R}^{M \times L} \).

- Compute the score vector \( \mathbf{w} \in \mathbb{R}^M \) by selecting the maximum value of each row in \( \mathbf{W} \).

- Discriminate potential anomalies in the score vector using the threshold \( \alpha \).

Remark: Test signals with high probability, given the learned HMM model, are very likely to belong to the normal set (to be normal).
Evaluation Metrics

▶ Confusion matrix

<table>
<thead>
<tr>
<th>Groundtruth</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>False Negative (FN)</td>
</tr>
<tr>
<td>Abnormal</td>
<td>False Postive (FP)</td>
</tr>
</tbody>
</table>

Metrics for Correct Detection

Precision \( = \frac{TP}{(TP+FP)} \)
Recall \( = \frac{TP}{(TP+FN)} \)
F-score \( = \frac{2TP}{(2TP+FP+FN)} \)

*Metrics close to 1 means a good performance

Methods to be compared

- OC-SVM [RBF Kernel]: One-Class Support Vector Machine

Synthetic Dataset

A set of Gaussian sequences of \( T = 600 \) temporal points was generated. The set is divided as:

- Learning set: \( N = 500 \)
- Testing set: \( M = 100 \)
Parameter setting

Remark: $X = [x_1, ..., x_n, ..., x_N]^T$ is the set of time series, and $x_n = [x_{n,1}, ..., x_{n,T}]$, with $x_n \in \mathbb{R}^T$.

Scenarios: For $u \neq 0$, $v \neq 1$, and $t_s$: temporal segment.

<table>
<thead>
<tr>
<th>Mean Value Changes</th>
<th>Variance Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Scenario 1</strong></td>
<td><strong>Scenario 3</strong></td>
</tr>
<tr>
<td>Learning</td>
<td>Learning</td>
</tr>
<tr>
<td>$H_0$</td>
<td>$H_0$</td>
</tr>
<tr>
<td>$x_n \sim \mathcal{N}(0, 1)$</td>
<td>$x_n \sim \mathcal{N}(0, 1)$</td>
</tr>
<tr>
<td>$H_1$</td>
<td>$H_1$</td>
</tr>
<tr>
<td>$x_{n,t_s} \sim \mathcal{N}(u, 1)$</td>
<td>$x_{n,t_s} \sim \mathcal{N}(0, v)$</td>
</tr>
</tbody>
</table>

| **Scenario 2**     | **Scenario 4**   |
| Learning           | Learning         |
| $H_0$              | $H_0$            |
| $x_{n,t_s} \sim \mathcal{N}(u, 1)$ | $x_{n,t_s} \sim \mathcal{N}(0, v)$ |
| $H_1$              | $H_1$            |
| $x_n \sim \mathcal{N}(0, 1)$ | $x_n \sim \mathcal{N}(0, 1)$ |
Results from Synthetic Data

- Emission probabilities: Gaussian distributions (1).
- Introduced anomalies: $u = 1.6$ and $v = 1.9$.
- Fraction of anomalies: 10% of the testing set.

Table 1: Summary of results from the synthetic dataset

<table>
<thead>
<tr>
<th></th>
<th>Mean Value Changes</th>
<th>Variance Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Scenario 1</td>
<td>Scenario 3</td>
</tr>
<tr>
<td></td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>OC-SVM</td>
<td>0.684</td>
<td>1.000</td>
</tr>
<tr>
<td>HMAD</td>
<td>0.833</td>
<td>0.666</td>
</tr>
<tr>
<td>HMM-Learn</td>
<td><strong>0.929</strong></td>
<td>1.000</td>
</tr>
</tbody>
</table>

|                          | Scenario 2         | Scenario 4       |
| OC-SVM                   | NaN*               | NaN              |
| HMAD                     | **1.000**          | **1.000**        | **1.000**| 0.067              | 0.143            | 0.090   |
| HMM-Learn                | **1.000**          | **0.909**        | **0.952**| **0.900**          | **1.000**        | **0.947**|

*NaN value: Anomalies not detected (TP = 0 and FP = 0).
Case of study: Temporal Remote Sensing

**Dataset** [Provided by Florian Mouret]

**Dataset Parameters:**

- **Area of study:** Beauce, France
- **Crops:** Rapeseed
- **Temporal resolution:** 13 images from the Sentinel-2 satellite, corresponding to the harvest season of rapeseed (October 2017 to June 2018).

**Time Series Extraction**

1. Parcel Extraction
2. NDVI estimation of each parcel
3. Median of NDVI

**Total:** 2218 parcels (Time series)
Anomalies in Rapeseed crops

Anomalies in the Dataset

1. Late/Early growth
2. Heterogeneity
3. Late/Early senescence
4. Early flowering
5. Wrong shape

Available labels (Total)

- Non-labeled: 1329
- Abnormal Labels (1): 725
- Normal Labels (0): 164

Illustration of normal and abnormal time series
Example of Anomalies in the Dataset

Wrong shape

Heterogeneity

Heterogeneity after senescence
Quantization

Given that the temporal resolution of the dataset is low \((T = 13)\) Gaussian distributions are not suitable. Thus, the set of time series is discretized as follows:

- **Linear binning (regular):** The width of bins is 0.1. The codebook has 10 numbers, \(K = 10\).

- **Non-linear binning:** Binning based on the median \(\mu_G\) and standard deviation \(\sigma_G\) of the normal sequences on each instant of time. The discretization can be written as

\[
h(x) = \begin{cases} 
    x_t = 1, & \text{if } x_t < \mu_G^t - 3\sigma_G^t \\
    x_t = 2, & \text{if } \mu_G^t - 3\sigma_G^t \leq x_t \leq \mu_G^t + 3\sigma_G^t \\
    x_t = 3, & \text{if } x_t > \mu_G^t + 3\sigma_G^t
\end{cases}
\]

for \(t = 1, \ldots, T\), where the number of elements in the codebook \(K = 3\).
Performance vs. Number of states

![Graphs showing performance vs. number of states for single and multiple HMM model selection with linear and non-linear binning.]

- **Linear binning**
  - Precision, Recall, and F-score for different number of states.
  - Graphs for single HMM model selection and multiple HMM model selection.

- **Non-Linear binning**
  - Similar to linear binning but with different performance metrics.
  - Graphs for single and multiple HMM model selection.
Results from the Real Dataset

Performance Comparison

Table 2: Results from the different detectors based on the data with available labels.

<table>
<thead>
<tr>
<th>Detector</th>
<th>Precision</th>
<th>Recall</th>
<th>F-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>OC-SVM</td>
<td>0.867</td>
<td>0.206</td>
<td>0.333</td>
</tr>
<tr>
<td>HMAD</td>
<td>0.873</td>
<td>0.492</td>
<td>0.629</td>
</tr>
<tr>
<td>HMM-Learn</td>
<td>0.894</td>
<td>0.724</td>
<td>0.799</td>
</tr>
</tbody>
</table>
The work is not finished! However, some conclusions are available:

- **A better performance in detection is obtained** in the anomaly detection approach based on HMM learning from time series.

- HMM learning for anomaly detection is **suitable for both high and low temporal resolution** sequences by properly selecting the distribution density.

- The proposed approach **allows the detection of anomalies related to mean value changes, variance changes, and even no changes**, as long as the learning step receives as input the signals considered as normal.

![Graphs showing Mean Value change, Variance change, and No change](image-url)
Future Work

Theory

► How to design the best HMM model selection in the current algorithm? (Minimum coherence, distance criterion).
► How to detect when occurs (time) the anomaly?
► To explore: Semi-Hidden Markov models (each state has variable duration), HMM states design.

Applications

► Spectral video and compressive spectral video sensing
Future Work

Applications

▶ Apply the approach to citrus crops in Colombia: The main limitation is that there is few information about normal and abnormal citrus.

▶ Citrus project (ECOS-Nord 2019): Anomaly detection in citrus using Optical and SAR data.

▶ **AgroTIC**: App for the agriculture
References

Anomaly Detection Surveys


Hidden Markov Models


Algorithms used in Experiments


▷ N. Görnitz, M. Braun, and M. Kloft, Hidden Markov anomaly detection, in International conference on machine learning, 2015, pp. 1833-1842.
AgroTIC Project

Thanks you for your attention!

Questions?
Merci!/Thanks!/¡Gracias!